Master 2 Atiam : Recherche par similarité musicale avec application à la similarité mélodique, harmonique et rythmique

Rémi Foucard

22 juin 2009

Glossaire

	Anglais	Français
DPLA	*Dynamic Programming Local Alignment*	*Alignement Local par Programmation Dynamique*
DTW	*Dynamic Time Warping*	*Déformation temporelle Dynamique*
FFT	*Fast Fourier Transform*	*Transformée de Fourier Rapide*
HPCP	*Harmonic Pitch Class Profile*	*Profil de Classes de Hauteurs Harmoniques*
OTF	*Optimal Transposition Factor*	*Facteur Optimal de Transposition*
OTI	*Optimal Transposition Index*	*Index Optimal de Transposition*

Table des matières

Introduction

Depuis quelques années, la taille grandissante des bibliothèques musicales appelle de nouveaux outils pour rechercher des morceaux de musique. Devant la quantité colossale de morceaux disponibles, l'utilisateur a besoin d'outils de recherche intuitifs et pertinents. Ces outils pourraient rechercher les fichiers en fonction de leur contenu, en plus d'informations extérieures telles que le nom du compositeur ou l'année d'enregistrement.

Le calcul de similarité musicale entre deux morceaux s'inscrit dans ce type d'applications : il pourrait permettre de rechercher dans une base, les morceaux similaires à une chanson donnée. Il est difficile de définir de manière absolue la notion de similarité musicale : elle peut se définir au niveau de l'harmonie, de la mélodie, du style, de l'humeur, etc. Afin comparer nos expérimentations à une vérité-terrain claire, nous avons choisi d'orienter nos travaux sur une sous-partie du concept de similarité musicale. Nous allons alors considérer que la requête idéale sur un morceau renverrait en premier les fichiers constituant une autre interprétation de la même œuvre. Notre approche de la similarité musicale est donc confondue avec l'identification de reprises (ou "*cover versions*").

De nombreux travaux ont déjà été effectués dans ce domaine, se basant sur la séquence tonale du morceau, ou bien sur sa mélodie. Dans nos travaux, nous essaierons de concevoir un système prenant en compte séparément la similarité au niveau des mélodies d'une part, et des séquences d'accords d'autre part. Plutôt que d'inventer un algorithme entièrement nouveau, nous préférerons nous baser sur une technique de calcul de similarité déjà existante. Nos travaux consisteront alors à adapter cet algorithme pour lui permettre de prendre en compte des données enrichies.

Après avoir donné une vue d'ensemble de l'état de l'art actuel de la recherche sur l'identification de reprises, nous décrirons plus précisément le système sur lequel nous avons basé nos recherches. Puis, nous détaillerons notre système d'analyse de similarité et ses caractéristiques. Nous verrons enfin comment nos recherches nous ont amenés à envisager plusieurs types de descripteurs pour les mélodies, et à mettre au point une gestion réfléchie des passages silencieux.

1 Retour sur la notion de "reprise"

Une reprise est la réinterprétation d'un morceau existant par un autre interprète que son créateur. On peut également parler de différentes "versions".

Une grande partie des reprises actuelles cherche à adapter, à transformer un morceau, pour lui donner une nouvelle dimension. Cependant, il est

toujours possible à un auditeur, à l'écoute d'une reprise, d'identifier la chanson originale s'il la connaît. On peut donc en déduire que quelques traits de l'œuvre originale ont été conservés.

On trouve parmi les modifications les plus notables entre une chanson originale et sa reprise :

les arrangements : ils constituent certainement le changement le plus courant. Les instruments présents ne sont pas les mêmes, une voix peut être remplacée par un instrument, ou l'inverse.

la tonalité principale : elle est souvent changée, par exemple pour s'adapter au registre du chanteur ;

le tempo : certaines reprises vont jusqu'à doubler le tempo de l'originale ;

le rythme : notamment dans le jazz, les interprètes aiment apporter leur empreinte rythmique, leur *swing* personnel, et ainsi modifier le rythme de la mélodie. En outre, dans le cadre d'un changement de style, un morceau peut se voir reconstruit sur un rythme qui ne lui correspondait pas au départ (par exemple, en changeant le type de mesure, ou en reprenant un morceau rock sur un rythme de samba).

la structure du morceau : ajout ou suppression d'une introduction ou d'une transition, répétition d'une partie, etc.

la progression d'accords : elle peut être modifiée par des substitutions, voire totalement remplacée dans certains cas ;

la mélodie principale : dans les reprises de morceaux jazz, la progression d'accords reste en général relativement stable, tandis que les solistes improvisent la mélodie.

Il semble donc difficile de trouver un invariant unique, partagé par toutes les reprises d'une même composition. Cependant, il est rare que tous ces aspects du morceau soient modifiés dans une même reprise. Ainsi, la progression tonale (qui comprend la mélodie et les accords) reste relativement stable.

2 État de l'art au début du stage

Les méthodes de calcul de similarité possèdent deux caractéristiques principales, correspondant aux deux grandes étapes dans leur procédure (figure 1) :

- leur choix de descripteurs à extraire des morceaux ;
- les techniques utilisées pour comparer et aligner temporellement ces descripteurs.

Les descripteurs qui nous intéressent sont basés principalement sur des séquences tonales, ainsi que sur des techniques d'extraction de la mélodie principale. Les méthodes pour comparer les descripteurs peuvent ensuite être

classées en deux catégories : l'approche corrélative, et les techniques basées sur la *Dynamic Time Warping* (DTW).

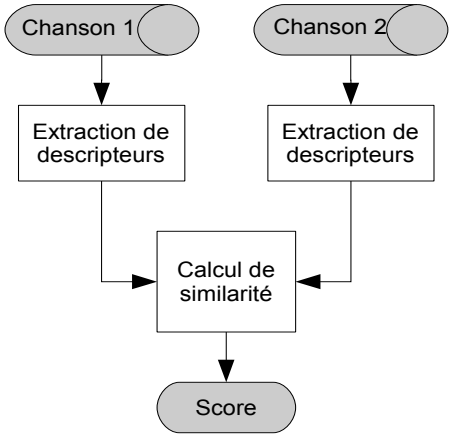

FIG. 1 – Organisation globale du calcul de similarité musicale

2.1 Extraction des descripteurs

Comme nous l'avons vu dans la partie 1, la séquence tonale d'un morceau constitue une des caractéristiques qui varient le moins parmi les différentes interprétations d'un morceau. C'est pourquoi les chroma, qui représentent cette séquence, sont largement utilisés pour l'identification de reprises. On trouve aussi de nombreux travaux basés sur l'extraction de mélodie principale.

2.1.1 Descripteurs tonaux : Séquences de chroma

Un vecteur de chroma représente l'énergie spectrale du signal à travers les 12 demi-tons d'une octave, mais ne fait pas de différence entre deux octaves distinctes. Les chroma apparaissent donc comme des classes de hauteurs. Il est également possible d'adopter des vecteurs de résolution supérieure à un demi-ton (vecteurs de 24 ou 36 chroma, par exemple).

Les descripteurs basés sur les chroma sont assez répandus pour les applications de calcul de similarité, et fonctionnent apparemment assez bien [7], [4], [14]. Parmi les intérêts des chroma, on peut citer leur robustesse face au bruit, leur indépendance du timbre et des instruments qui jouent, ainsi que de l'intensité sonore.

On trouve plusieurs sortes de chroma dans la littérature, qui se singularisent par la manière de les calculer. Leur construction commence en général par une analyse temps-fréquence du signal audio par Transformée de Fourier à Court Terme (on obtient ainsi un vecteur de chroma pour chaque trame

d'analyse de Fourier). Ensuite, le calcul des vecteurs suit différentes procédures.

Les *Pitch Class Profiles* [5] sont obtenus en sommant, les points de la TFCT en fonction du chroma correspondant à leur fréquence associée. Ces descripteurs ont été améliorés par [15] et constituent une référence pour la plupart des autres systèmes.

Les *CQ-profiles* [10] proposent de calculer les vecteurs de chroma en décomposant le signal par un banc de filtres à Q constant (Q étant le rapport entre la fréquence centrale de chaque filtre, et la largeur de sa bande passante). Ces filtres sont centrés sur les demi-tons de la gamme chromatique. Ainsi, pour obtenir les coefficients *CQ-profiles*, on somme les composantes du signal correspondant à chaque chroma.

La méthode décrite dans [6] consiste à extraire du signal les pics spectraux, puis à les sommer en fonction de leur chroma correspondant. Puisque des pics spectraux apparaissent généralement aux fréquences harmoniques d'une fondamentale, chaque pic extrait contribuera, non seulement à son chroma correspondant, mais également aux chroma correspondant à ses fréquences sous-harmoniques. Notons également que ces chroma, appelés *Harmonic Pitch Class Profile* (HPCP) s'adaptent au diapason utilisé dans le morceau. Nous décrirons plus précisément les HPCP dans la section 3.2.2.

La figure 2 montre une séquence de chroma calculée à partir d'un signal de piano monophonique de six secondes. Cette séquence a été calculée par la méthode des HPCP, dont la résolution a été portée à 120 classes de hauteurs pour la visibilité du graphique. On peut bien distinguer en noir les notes jouées par le piano, même si quelques résidus apparaissent en gris, correspondant aux fréquences harmoniques.

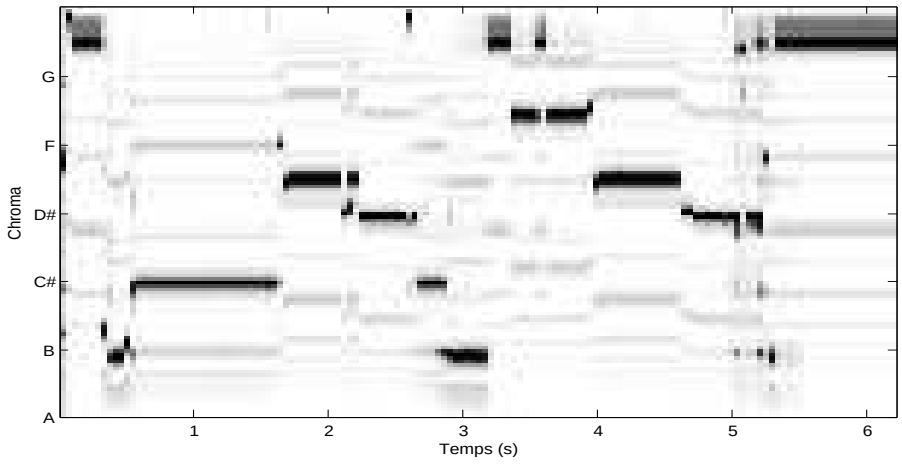

FIG. 2 – Séquence de HPCP calculée à partir d'un signal monophonique

2.1.2 Extraction de la mélodie principale

Beaucoup considèrent que la mélodie principale constitue l'essence d'un morceau de musique, sa spécificité. Si l'on poursuit cette logique, c'est donc elle qu'il faut comparer afin de calculer la similarité entre deux morceaux. On trouve ainsi de nombreux articles se fondant sur cette idée intuitive.

Les techniques d'extraction sont variées et leur résultat peut se trouver sous forme symbolique [16], intermédiaire [9] ou audio [2]. La similarité de ces mélodies pourra être calculée de la même manière que pour les séquences de chroma mais il existe également des techniques spécifiques [9], [16], [17]. Notamment, pour des mélodies sous format symbolique, on trouve des techniques dérivées de l'alignement de chaînes de caractères.

2.2 Calcul de similarité

Une fois extraits les descripteurs que l'on juge pertinents, il faut chercher à comparer leurs séquences de manière appropriée, c'est à dire qui représente bien la similarité musicale. La plupart des méthodes de comparaison se basent soit sur une approche corrélative, soit sur des alignements temporels de type *Dynamic Time Warping* (DTW).

2.2.1 Corrélation croisée

Cette approche consiste à mesurer la similarité des deux séquences considérées, en calculant directement leur corrélation.

Les valeurs de corrélation sont ensuite normalisées en fonction de la séquence la plus courte, de manière à obtenir des mesures entre 0 et 1.

Dans [4], les auteurs remarquent que les morceaux représentant la même chanson sont non seulement indiqués par des valeurs de corrélation élevées, mais également par le fait que ces valeurs forment des pics très étroits autour de l'alignement optimal. Puisque les valeurs de corrélation diminuent rapidement autour de cet alignement, la corrélation est donc filtrée passe-haut, de manière à mettre en évidence les pics étroits.

Suivi de tempo Pour calculer la corrélation entre deux vecteurs, il est nécessaire qu'ils aient une échelle temporelle similaire. Comme le souligne [9], le temps en secondes ne semble pas être une échelle appropriée, puisqu'un morceau et sa reprise peuvent avoir des tempi différents. C'est pourquoi la plupart des techniques utilisant la corrélation, choisissent la pulsation rythmique comme unité temporelle dans leurs représentations [9], [4], [14]. Elles nécessitent donc l'utilisation d'un algorithme de suivi de tempo.

La corrélation a l'avantage d'être peu coûteuse et de bien représenter l'alignement global des morceaux, mais nécessite l'utilisation d'un suivi de pulsation rythmique afin de pallier aux différences de tempo.

2.2.2 Méthodes basées sur la *Dynamic Time Warping*

Une autre manière de gérer les différences et fluctuations de tempo est proposée par la DTW. Cette technique a pour avantage de s'affranchir des méthodes de suivi de pulsation, puisqu'elle a été conçue spécialement pour gérer les variations temporelles. Pour comparer deux séquences, la DTW permet de les déformer temporellement pour tenter d'aligner ensemble leurs caractéristiques similaires.

Description de la DTW Considérons deux séquences Q et C, de tailles respectives n et m. On construit à partir de ces séquences une matrice D, de taille $n \times m$, qui définit la distance cumulative $D(i, j)$, comme étant la distance locale $d(Q(i), C(j))$ (`cf.` section 2.2.3), ajoutée au minimum des distances cumulatives des éléments adjacents :

$$D(i, j) = d(Q(i), C(j)) + \min\{D(i-1, j-1), D(i-1, j), D(i, j-1)\}$$

Cette opération permet de trouver le coût total d'alignement (élément (n, m) de la matrice), ainsi qu'un chemin d'alignement (figure 3), dont la longueur constitue un facteur de normalisation.

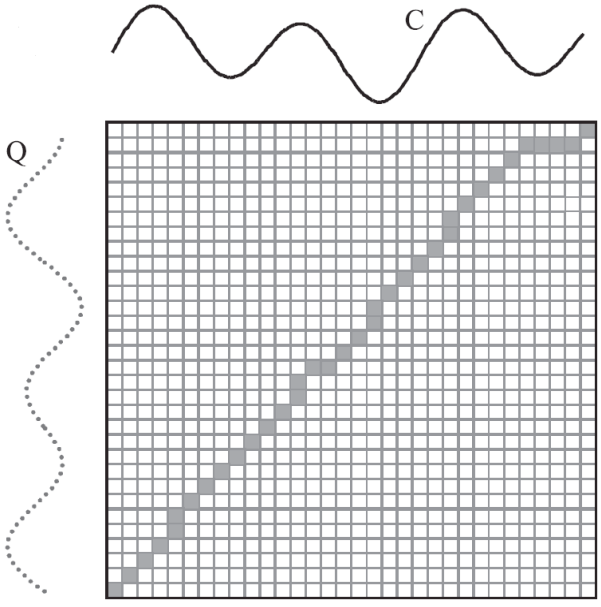

FIG. 3 – Chemin d'alignement de deux séquences par DTW

La DTW est largement utilisée depuis plusieurs décennies pour le traitement de la parole. Ainsi, on en trouve de nombreuses descriptions détaillées [8], [11].

Alignement local par programmation dynamique Si la DTW permet de trouver un alignement global raisonnable, elle n'est en revanche pas conçue pour détecter des alignements locaux de sous-séquences au sein des morceaux comparés. Pourtant, nous l'avons vu dans la section 1, beaucoup de reprises modifient la structure de la chanson originale. Partant de ce constat, l'algorithme proposé dans [14], nommé *Dynamic Programming Local Alignement* (ou DPLA), cherche à aligner des sous-séquences. Il sera décrit dans la section 3.2.3.

2.2.3 Paramètres communs des calculs de similarité

Les calculs de similarité entre morceaux quels qu'ils soient, doivent faire face à des problèmes communs : définir une mesure de distance locale entre deux descripteurs extraits (section 2.1), et gérer les différences de tonalité entre les morceaux.

Mesure de distance locale Toutes les mesures de similarité globale nécessitent de comparer des instants précis de chaque morceau. Pour les vecteurs de chroma, on peut utiliser la distance euclidienne, de corrélation, ou cosinusoïdale.

Les résultats de ces comparaisons sont variables mais aucune mesure ne semble significativement plus appropriée. La distance perceptive entre les chroma est encore à l'étude, c'est pourquoi [14] utilise une similarité binaire : "*similaire*" ou "*pas similaire*".

Pour les mélodies, on peut, comme pour les chroma, calculer une distance cosinusoïdale [9], ou prendre la valeur absolue de la distance entre les deux notes [16].

Les mesures de distance locale sont donc très diverses et fonctionnent plus ou moins bien selon les applications. Pour l'instant, aucune n'apparaît comme significativement meilleure que les autres. Il paraît donc nécessaire de tester leur effet en fonction des applications.

Prise en compte des différences de tonalité Un morceau et sa reprise peuvent avoir des tonalités différentes (cf. section 1). Cependant, afin de comparer deux séquences de manière appropriée, il est nécessaire qu'elles soient dans la même tonalité.

Certains choisissent de calculer douze fois la distance entre les morceaux, en transposant l'un d'entre eux dans toutes les tonalités. La plus petite distance sera choisie. Une autre approche consiste à estimer la tonalité des deux morceaux, et transposer l'un dans la tonalité de l'autre. Enfin, on trouve dans [14] une technique pour trouver un *Index Optimal de Transposition* (OTI, en anglais) sans passer par une estimation de tonalité. Le calcul de cet OTI sera détaillé page 15.

3 Cadre des travaux effectués et description des tests

Nous avons choisi de baser nos travaux sur un système dejà existant, sans en remettre en cause le principe de fonctionnement. Nos expérimentations ont consisté à étendre cet algorithme et/ou à le faire tourner sur des entrées plus riches que le simple signal audio du morceau. L'évaluation de nos travaux a été réalisée selon la procédure décrite ci-après.

3.1 Procédure de test et valeurs observées

Les performances de nos algorithmes ont été testées en effectuant des requêtes sur une base de données. Chaque requête a pour contenu une chanson, et les morceaux de la base sont alors retournés par ordre de similarité décroissante avec cette chanson.

Pour faire ces requêtes, nous avons utilisé la base Covers80[1], proposée par D. Ellis. Cet ensemble de 164 morceaux comporte 80 chansons pop accompagnées, en général, d'une reprise. Les morceaux sont échantillonnés en mono à 16 kHz, avec des échantillons de 16 bits. Nous avons également contribué à l'établissement d'une nouvelle base de 1000 morceaux, comprenant 20 versions pour chaque composition. Cette construction semble davantage adaptée au test d'une application de recherche de morceaux, mais nous n'avons malheureusement pas encore pu l'utiliser pour nos évaluations.

Etant donné que Covers80 ne fournit qu'une seule reprise par morceau original, les mesures de rappel, précision et F-mesure[2], préconisées par [13], nous semblent insuffisantes pour rendre compte de la pertinence du retour d'une requête. En effet, si la réponse à une requête ne comporte qu'un seul morceau, ces mesures déterminent uniquement si la reprise a été retournée en premier ou non. En revanche, si l'on retourne tous les morceaux sauf celui de la requête, le rang de la reprise dans la liste retournée (ou de la première reprise, s'il y en a plusieurs), nous semble un indicateur plus parlant. On pourra calculer sa moyenne sur un ensemble de requêtes, afin d'avoir un indicateur plus général.

Par ailleurs, en effectuant des requêtes successivement à partir des 164 morceaux de la base, nous avons observé que le rang de la reprise présentait un écart-type très élevé. Beaucoup de requêtes renvoient la reprise dans les

[1] La base de données Covers80 est décrite sur sa page dédiée, à l'adresse suivante : *http ://labrosa.ee.columbia.edu/projects/coversongs/covers80/*. Elle peut également être téléchargée depuis cette page.

[2] Ces mesures, très utilisées dans le domaine de la recherche d'information, permettent d'évaluer la pertinence d'une liste de documents retournée par une requête. La précision, par exemple, désigne le nombre de reprises retournées par rapport au nombre total de morceaux retournés.

tout premiers morceaux, et beaucoup donnent une réponse très mauvaise. Il y a, en fait, peu de résultats moyens. Nous avons, en outre, remarqué que sur les 164 requêtes, le rang médian de la reprise recherchée était souvent inférieur à 10. En observant la variation de cette médiane lors de nos expérimentations, nous pouvons ainsi observer les performances de notre système, en donnant une moindre importance aux variations des mauvais résultats. Nous accordons en effet plus d'importance au fait que le rang de la reprise passe, par exemple, de 9 à 5 dans une réponse, plutôt que de 99 à 95. Dans le premier cas, une bonne réponse devient très bonne, tandis que dans le second cas, on peut considérer que la reprise n'a pas été trouvée, qu'elle soit classée 99^{e} ou 95^{e}.

Nous retiendrons donc pour évaluer les performances de nos travaux : la précision moyenne, le rang moyen de la reprise (ou de la première reprise) et le rang médian. Notons que, pour le premier indicateur, des chiffres élevés indiquent de bonnes performances, tandis que dans les deux autres cas, c'est l'inverse.

3.2 Système de base utilisé

3.2.1 Choix du système

Nous avons choisi de baser nos expérimentations sur l'algorithme *Dynamic Programming Local Alignment* (DPLA) [14], pour deux principales raisons.

Premièrement, dans cet algorithme, la similarité locale calculée entre deux trames d'analyse, n'a que deux valeurs possibles. Le fait que cette similarité soit binaire facilite le traitement conjoint et la fusion de descripteurs de nature hétérogène. En effet, à un instant donné, lors de la comparaison de deux morceaux, différents descripteurs peuvent avoir des similarités variées. Le fait de n'avoir que deux valeurs possibles évite d'avoir à faire des analyses sémantiques complexes sur ces valeurs, puisque leur sens est déjà on ne peut plus clair : "*similaire*" ou "*pas similaire*".

Notre choix a également été porté sur le DPLA au vu des excellents scores obtenus au MIREX 2007[3] par cet algorithme, arrivé premier dans toutes les catégories. En basant notre travail sur un système qui s'est avéré aussi performant, nous avons donc de bonnes chances d'obtenir de meilleurs résultats.

Cet algorithme est basé sur une représentation des morceaux sous forme de HPCP. Nous expliquerons d'abord leur calcul avant de décrire le système DPLA proprement dit.

[3] http ://www.music-ir.org/mirex/2007/

3.2.2 Calcul des HPCP

Les HPCP sont des vecteurs de chroma basés sur les *Pitch Class Profiles* [5]. Leur méthode de construction est décrite en détails dans [6], et résumée dans la figure 4.

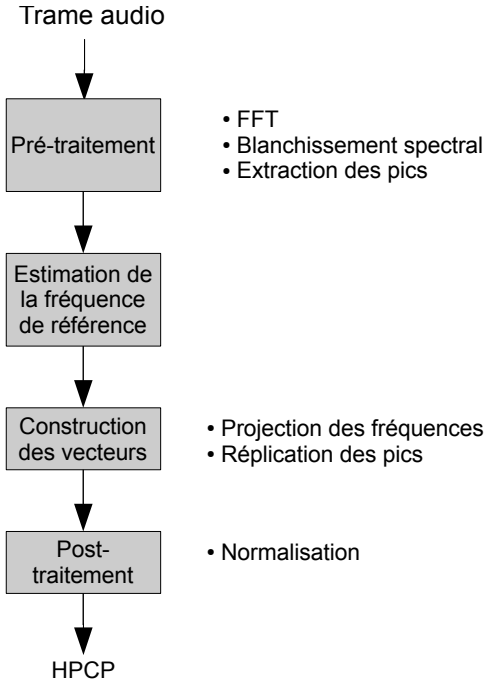

FIG. 4 – Représentation globale du calcul d'une séquence de HPCP

Pré-traitement Cette méthode commence par une analyse temps-fréquence du signal. Chaque trame de 4096 échantillons sera multipliée par une fenêtre de *Blackman Harris 62 dB*. Le début d'une trame se situe 512 échantillons après le début de la précédente. On va ensuite appliquer une FFT sur chacune des trames ainsi définies, et normaliser le spectre obtenu par son amplitude. Cette opération permet d'être moins sensible au timbre des instruments qui jouent. Puis, on extraira les pics (maximums locaux) du spectre ainsi blanchi. Les mesures de fréquence et d'amplitude de chaque pic seront affinées par interpolation quadratique. Pour la suite des opérations, on gardera uniquement les pics :

- dont la fréquence se situe dans l'intervalle [100, 5000] Hz ;
- dont l'amplitude est supérieure à -100 dB par rapport à l'amplitude maximale possible.

Estimation de la fréquence de référence Tous les morceaux ne sont pas accordés sur un La à 440 Hz. Afin d'adapter nos classes de hauteurs à

la fréquence de référence du morceau en cours d'analyse, on va formuler une estimation de cette fréquence.

Pour cela, nous allons projeter les pics extraits sur une échelle de demi-tons accordés à 440 Hz. On va alors pouvoir mesurer la déviation dev_i, en demi-tons, entre la fréquence de chaque pic i et la fréquence du demi-ton le plus proche. On a donc $dev_i \in [-0.5, 0.5]$. Pour chaque trame d'analyse spectrale, on va sommer l'énergie des pics dans un histogramme en fonction de leur déviation dev_i :

$$hist(n) = \sum_{i,\ dev_i+0.5 \in [(n-1)r, nr]} a_i$$

où $n = 1, 2, \ldots$, où r est la résolution en demi-tons, et a_i est l'amplitude linéaire du pic i.

Cette opération donne un histogramme des déviations pour chaque trame. La déviation locale sera estimée au maximum de chaque histogramme. Puis, de la même manière, on va sommer l'énergie des trames dans un histogramme global en fonction de leur déviation. La déviation estimée pour tout le morceau correspondra au maximum de l'histogramme global. On pourra alors en déduire f_{ref}, la fréquence de référence du morceau.

Construction des vecteurs de chroma Pour chaque trame d'analyse, l'énergie d'un point n du HPCP est définie par :

$$HPCP(n) = \sum_{i=1}^{nPeaks} w(n, f_i) \cdot a_i^2 \qquad n = 1 \ldots size$$

où a_i et f_i sont la fréquence et l'amplitude linéaires du pic i et $w(n, f_i)$ est une fenêtre autour de la fréquence centrale f_n du point n du HPCP. La fréquence f_n est calculée de la manière suivante :

$$f_n = f_{\text{ref}} \cdot 2^{\frac{n}{size}}$$

Grâce à la fenêtre w, chaque pic ne contribue pas à un seul chroma, mais à tous les chroma dont f_i, la fréquence du pic, est proche. Cette fenêtre possède une largeur de $4/3$ demi-ton et sa forme est en $\cos^2$ autour de la fréquence centrale du chroma, f_n (l'allure de w est présentée dans la figure 5). Les valeurs de w sont fonction de d, la distance en demi-tons entre la fréquence du pic f_i et la fréquence centrale du chroma :

$$d = 12 \cdot \log_2 \left(\frac{f_i}{f_n} \right) + 12m$$

où m est l'entier qui minimise le module de la distance $|d|$.

Afin de considérer la présence de pics spectraux aux fréquences harmoniques des fondamentales, chaque pic sera répliqué aux premiers sous-multiples de sa fréquence. Ces répliques auront une amplitude de moins en moins élevée.

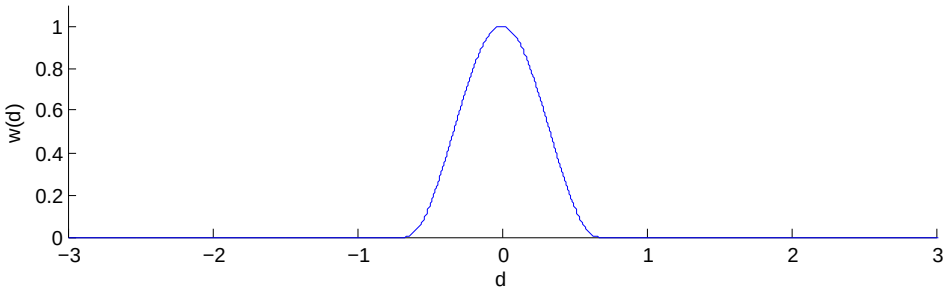

FIG. 5 – Amplitude de la fenêtre w en fonction de la distance d, en demi-tons, entre la fréquence du pic et la fréquence centrale du chroma considéré

Post-traitement Pour rendre nos HPCP indépendants du volume sonore, on normalise, pour chaque trame, le HPCP correspondant par sa valeur maximale.

3.2.3 Matrice de similarité et alignement par DPLA

Comme nous l'avons expliqué dans la section 2.2.2, la technique *Dynamic Programming Local Alignment* (DPLA) est proche de la DTW mais est conçue pour aligner des sous-séquences, plutôt que de chercher un alignement pour le morceau entier. Cela permet de pallier aux différences de structure entre les morceaux. L'organisation globale de l'algorithme est présentée figure 6.

Pré-traitement Après avoir calculé, pour chaque morceau, une séquence de HPCP de 36 valeurs, on raccourcit cette séquence en remplaçant chaque groupe de 10 HPCP consécutifs, par leur moyenne. Cela permet d'accélérer la suite des calculs.

Afin de transposer le deuxième morceau dans la tonalité du premier, on va chercher leur *Index Optimal de Transposition*. Pour cela, on commence moyenner temporellement les deux séquences afin d'obtenir un HPCP global pour chaque morceau : $\overrightarrow{h_1}$ et $\overrightarrow{h_2}$. Puis, l'OTI sera donné par la formule :

$$OTI\left(\overrightarrow{h_1}, \overrightarrow{h_2}\right) = \text{argmax}_{0 \leq id \leq size-1} \left\{ \overrightarrow{h_1} \cdot \text{circshift}_R(\overrightarrow{h_2}, id) \right\}$$

où "$\cdot$" indique un produit scalaire, *size* est la taille des HPCP et circshift$_R(\overrightarrow{h}$, $n)$ est une fonction qui effectue une rotation du vecteur h, de n cases vers la droite.

Une fois l'OTI déterminé, on peut transposer $\overrightarrow{h_2}$ dans la tonalité de $\overrightarrow{h_1}$.

Matrice de similarité L'étape suivante consiste à construire une matrice de similarité entre les deux séquences. Comme nous l'avons dit, dans cette

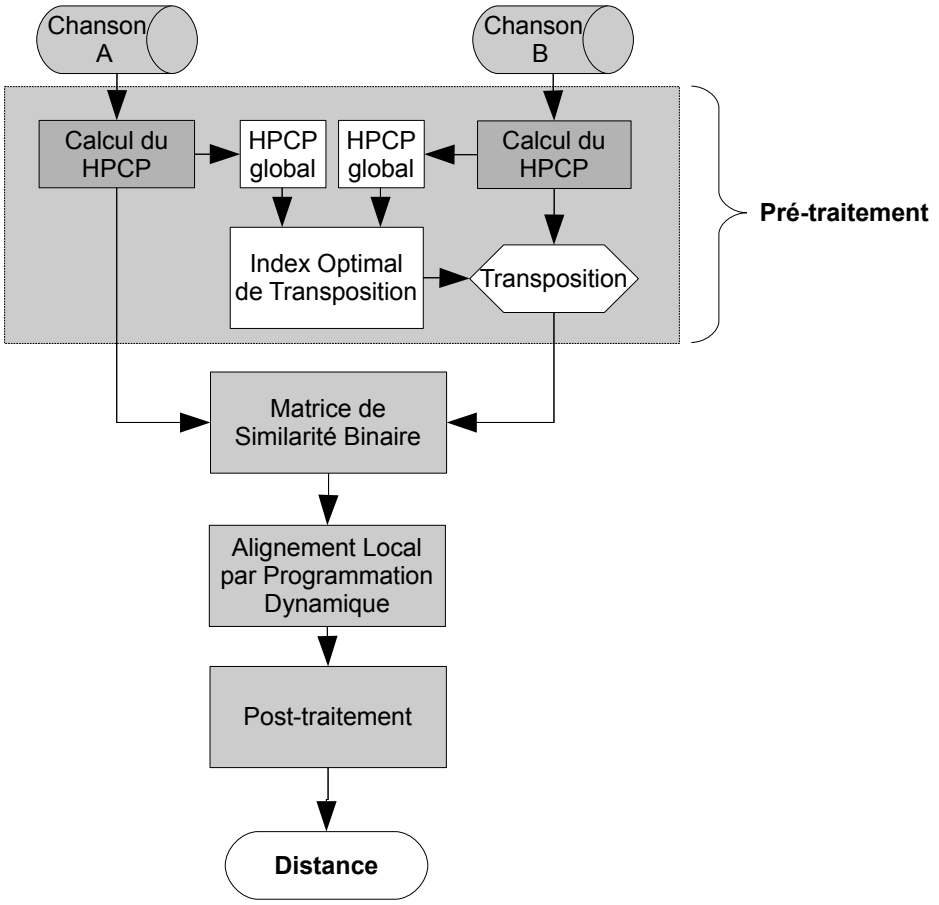

FIG. 6 – Organisation globale de l'algorithme *Dynamic Programming Local Alignment* (DPLA)

matrice, la similarité locale entre chaque paire de HPCP, est binaire. Elle n'a que deux valeurs possibles. Sa valeur dépend de l'OTI local calculé entre les deux vecteurs :

- si l'OTI est compris dans un intervalle d'un demi-ton autour de 0 (pour des HPCP de 36 valeurs, cela correspond à $OTI \in \{0, 1, 36\}$), alors les HPCP sont similaires ;
- sinon, les HPCP sont considérés non similaires.

Alignement par DPLA Cette matrice de similarité permet de construire une matrice d'alignement. A la différence de la DTW, cette matrice H ne calcule pas une distance cumulative globale, mais la similarité cumulative de sous-séquences. Dans cette matrice, chaque coefficient $H_{i,j}$ représente la similarité de deux sous-séquences se terminant respectivement en i et j. Ces scores augmentent au fil des sous-séquences similaires, puis diminuent

si la similarité locale est faible. Le score de similarité que l'on prendra sera égal au coefficient maximum de la matrice d'alignement : on considère qu'il représente la similarité cumulative des sous-séquences les mieux alignées. Cette valeur sera normalisée par la longueur des deux morceaux.

Un exemple d'alignement par DPLA est présenté en figure 7. Dans cette figure, les deux axes représentent le temps des deux morceaux. Les scores d'alignement les plus élevés, en rouge, indiquent les séquences similaires.

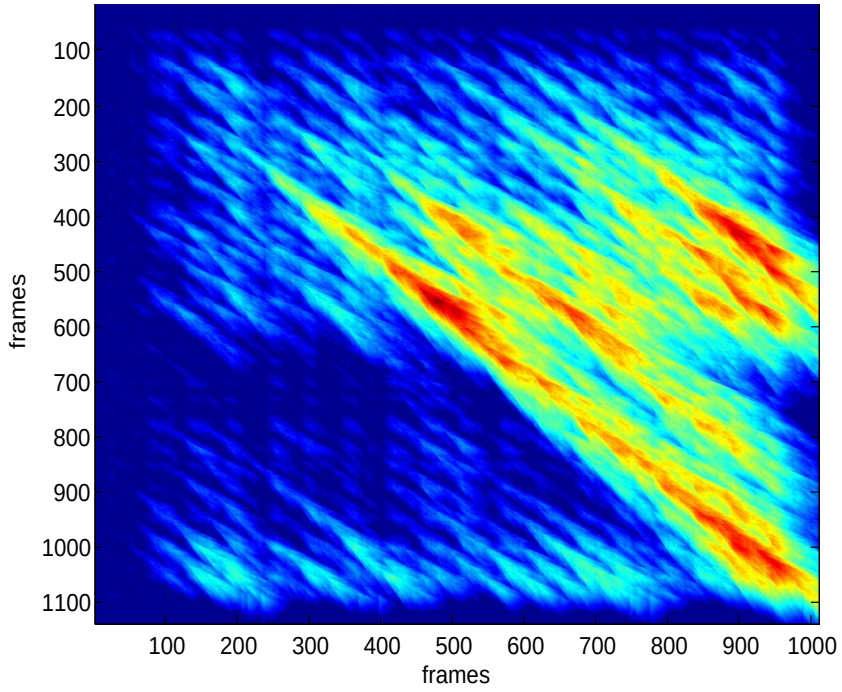

FIG. 7 – Alignement de deux versions de la chanson *Addicted to love* (par Tina Turner et Robert Palmer)

3.2.4 Performances de notre implémentation

Après avoir implémenté HPCP et DPLA, nous avons cherché à tester les performances de notre code.

Si nous n'avons pas pu évaluer notre implémentation du DPLA par rapport à une autre, nous avons, en revanche, pu comparer nos HPCP à l'implémentation fournie dans la MIRtoolbox[4].

Cette boîte à outils fournit des HPCP de 12 valeurs. C'est pourquoi nous avons, pour ces seuls tests, diminué la résolution de nos propres HPCP,

[4]La MIRtoolbox est disponible à l'adresse suivante :
http ://www.jyu.fi/hum/laitokset/musiikki/en/research/coe/materials/mirtoolbox

à 12 également. Nous avons ensuite comparé les performances du DPLA en utilisant soit les HPCP de la MIRtoolbox, soit notre implémentation (tableau 1). Notre implémentation semble bonne puisqu'elle conduit même à de meilleurs résultats.

	Précision moyenne	Rang moyen	Rang médian
Télécom Paristech	**0,33**	**37,50**	**13**
MIRtoolbox	0,27	40,49	16,5

Tab. 1 – Performances de l'algorithme DPLA pour deux implémentations d'HPCP de 12 valeurs

4 Séparation de la mélodie et analyse multimodale

4.1 Motivation

Nous avons vu dans la section 1 que la mélodie principale et les accords d'un morceau pouvaient se voir retouchés dans une reprise. Cependant, deux morceaux qui ne comportent ni la même mélodie ni la même progression harmonique, seront en général considérés par les auditeurs comme venant de deux compositions différentes. Mélodie et accords semblent donc essentiels pour l'identification d'un morceau.

Plusieurs études ont montré l'importance de la mélodie dans les tâches d'identification de morceaux par des auditeurs ([12], [1]). Mais il faut aussi prendre en compte certains cas où des reprises se basent sur les accords, souvent facilement reconnaissables, du morceau original, pour orner, broder et varier profondément le solo ; à tel point que le résultat n'a parfois plus grand chose à voir avec la mélodie d'origine. Cette pratique est courante en jazz.

La totalité des travaux de recherche sur la similarité musicale décrits dans la section 2 se base soit sur des représentations de type chroma, décrivant la progression tonale du morceau, soit sur des descripteurs symboliques ou intermédiaires de la mélodie, extraite en pré-traitement. Nous avons, pour notre part, tenté de concevoir un système prenant en compte séparément l'accompagnement et la mélodie, les considérant ainsi comme deux composantes du morceau.

4.2 Analyses simples

Le système le plus simple qui traite séparément solo et accompagnement, ne traite en fait qu'une seule de ces deux composantes.

4.2.1 Premiers tests

Avant toute chose, la mélodie et l'accompagnement ont été séparés au moyen de l'algorithme décrit dans [3]. Cet algorithme donne deux signaux audio pour chaque morceau analysé : l'un pour la mélodie, et l'autre pour l'accompagnement. Nous avons ensuite pu tester les performances de l'algorithme DPLA en prenant successivement comme signal de départ : le morceau brut, sans séparation, l'accompagnement, et la mélodie principale. Les tests ont été réalisés selon la procédure décrite dans la section 3.1. Ces analyses nous permettent de vérifier si le DPLA est approprié pour comparer les accompagnements ou les mélodies seules. Dans le cas contraire, il serait nécessaire de revoir le cadre de nos travaux.

Ces tests permettent également de voir, par exemple, si les différentes ornementations et variations de la mélodie principale, ne font pas baisser les performances de l'analyse de similarité. Dans ce cas, l'accompagnement seul mènerait à de meilleurs résultats que le morceau brut, sans séparation. Les résultats de ces tests sont résumés dans le tableau 2. Notons que la première ligne de ce tableau correspond à l'état de l'art que nous considérons, c'est à dire aux résultats du système dont nous sommes partis.

	Précision moyenne	Rang moyen	Rang médian
Brut (État de l'art)	**0,35**	**32,02**	**7**
Accompagnement	0,34	33,18	**7**
Solo	0,20	50,18	34

Tab. 2 – Résultats des analyses simples

Le premier constat est le suivant : ces résultats ne sont pas aberrants, l'algorithme de séparation de sources semble fonctionner correctement (ce qui est confirmé par l'écoute des mélodies et accompagnements seuls). On peut ensuite apporter deux commentaires :

- l'analyse de similarité sur l'accompagnement seul donne des résultats comparables à celle sur le morceau brut, bien que légèrement inférieurs ;
- en revanche, les requêtes effectuées sur les mélodies seules donnent des résultats bien moins bons.

Il ne semble donc pas raisonnable de comparer les solos de la même manière que les accompagnements ou les morceaux bruts.

4.2.2 Croisement des *Index Optimaux de Transposition*

Afin de comprendre et d'améliorer les mauvaises performances des comparaisons sur les solos, nous avons effectué une analyse d'erreur. Au cours de cette étude, nous nous sommes aperçus que dans beaucoup de cas, les requêtes donnant de mauvais résultats étaient dues à une mauvaise transposition des morceaux.

En effet, il arrive que certaines variations de la mélodie, entre le morceau original et sa reprise, conduisent à un mauvais calcul d'OTI. Les morceaux sont alors transposés dans des tonalités inadaptées et le DPLA n'aligne pas bien leurs séquences. Les deux versions du même morceau se retrouvent donc avec un score de similarité anormalement faible. Ainsi, lorsque l'on fera sur la base une requête contenant l'une de ces deux versions, l'autre se retrouvera dans les derniers résultats.

Pour remédier à ce problème, nous avons essayé de précalculer une table d'OTI, soit avec les morceaux bruts, soit avec les accompagnements. Cette table donne, pour chaque couple de morceaux, l'OTI correspondant. Nous avons ensuite comparé les mélodies en utilisant ces OTI précalculés. Afin de voir si cela améliorait les performances, nous avons aussi comparé les accompagnements en utilisant l'OTI calculé sur les morceaux bruts, et inversement.

	OTI	Précision moyenne	Rang moyen	Rang médian
Brut	Brut	**0,35**	**32,02**	**7**
	Acc^t	0,30	42,04	16,5
Accomp.	Brut	0,29	38,37	14
	Acc^t	**0,34**	**33,18**	**7**
Solo	Brut	**0,20**	**43,70**	**30**
	Acc^t	0,17	53,72	36
	Solo	**0,20**	50,18	34

Tab. 3 – Performances des analyses monocomposantes avec OTI croisés

Les résultats de ces tests sont donnés dans le tableau 3. On peut en tirer deux observations :

1. le croisement des OTI entre les morceaux sans séparation d'une part, et les accompagnements d'autre part, détériore les performances ;
2. les résultats des solos sont bien meilleurs si on utilise l'OTI calculé avec les morceaux bruts.

Puisque cela conduit à de meilleures performances, nous utiliserons donc dans tous nos tests ultérieurs, un OTI précalculé sur les morceaux bruts, lorsque nous comparerons les solos. Cependant, même avec un OTI précalculé, les performances ne sont toujours pas satisfaisantes. À cela, nous invoquons deux raisons possibles :

- les HPCP ne sont peut-être pas appropriés pour représenter une mélodie (ce point sera discuté dans la section 4.5) ;
- bien que la mélodie principale soit correctement extraite de chaque chanson lorsqu'elle est présente, quand le chanteur se tait en revanche, une partie de l'accompagnement est souvent considérée comme mélodie principale par notre algorithme d'extraction, ce qui peut fausser

les comparaisons (nous tenterons de remédier à ce problème dans la section 5.2).

Nous allons maintenant tenter de mettre en place des comparaisons de morceaux prenant en compte les deux composantes à la fois.

4.3 Description et paramètres du système multimodal

Après avoir effectué des tests sur une seule composante, nous avons réalisé un système permettant de calculer la similarité de deux morceaux en analysant séparément à la fois l'accompagnement et la mélodie principale. Sa procédure est présentée dans la figure 8.

On commence par extraire la mélodie et l'accompagnement au moyen de l'algorithme décrit dans [2]. Puis, on va calculer les descripteurs pour les deux mélodies et les deux accompagnements. Une fois transposés de manière appropriée, leur matrice de similarité binaire sera calculée, pour finalement effectuer un DPLA afin de donner un score de similarité entre les deux morceaux.

Deux caractéristiques de ce système vont varier dans nos expérimentations :

Fusion des analyses Le traitement parallèle de la mélodie et de l'accompagnement n'a d'intérêt que si, à un moment donné, les analyses sont jointes. Le stade où va s'effectuer cette jointure peut se trouver à divers endroits de l'algorithme. Et pour chaque endroit, plusieurs calculs sont possibles pour réaliser cette fusion. Les différentes jointures que nous avons envisagées sont présentées dans la section 4.4.

Types de descripteurs pour la mélodie Un solo et un accompagnement ne représentent pas le même type d'informations. Si, pour un accompagnement, la pertinence des HPCP comme descripteurs a été montrée dans la partie 4.2, il est beaucoup moins évident de trouver un type de descripteurs approprié pour la mélodie. Nous discuterons de ce point dans la section 4.5.

4.4 Point de fusion des analyses

Comme nous l'avons déjà expliqué, le traitement séparé du solo et de l'accompagnement appelle à joindre, à un moment ou à un autre, les analyses réalisées. Nous avons envisagé deux points possibles pour effectuer cette fusion : à la toute fin de l'algorithme, ou bien au niveau des matrices de similarité.

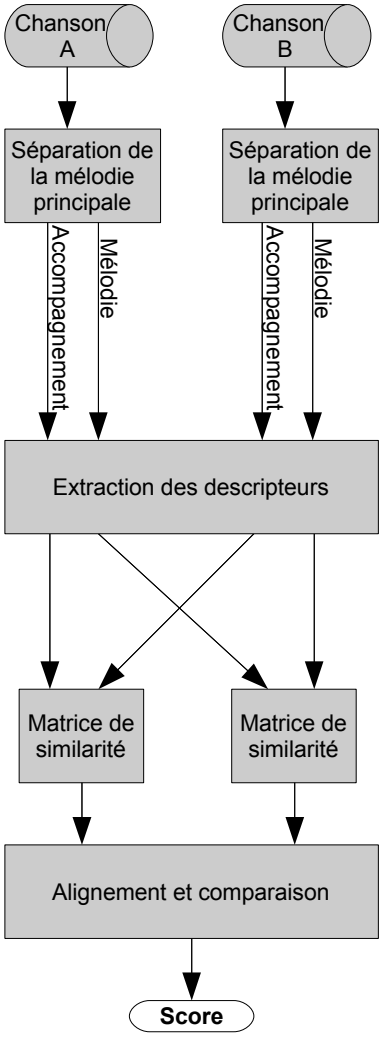

FIG. 8 – Principe du système d'analyse multimodale

4.4.1 Analyses entièrement parallèles, fusion à la fin

Dans cette configuration (figure 9), on effectue en fait deux analyses complètes, avec deux alignements par DPLA. Ces deux analyses donnent des scores de similarité, que l'on va combiner afin d'obtenir un score général pour les deux morceaux.

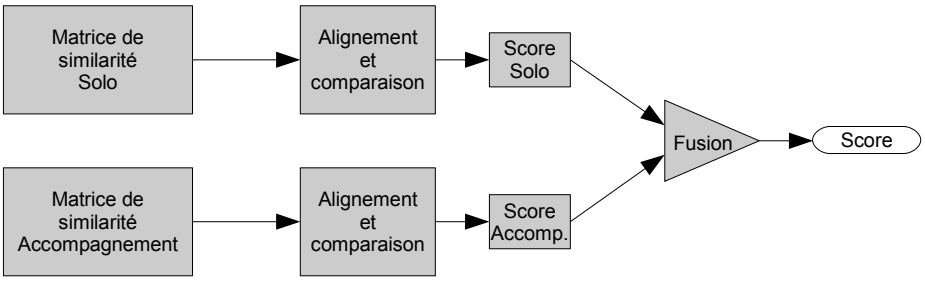

FIG. 9 – Fusion à la fin de l'analyse multimodale

Nous avons testé trois opérations pour combiner ces scores de similarité : l'addition, le minimum et le maximum. Ces opérations ont des sémantiques différentes au niveau de l'algorithme. Additionner les scores signifie en quelque sorte influencer l'analyse sur une composante, par l'analyse effectuée sur l'autre composante. Cela équivaut à prendre la moyenne. Prendre le minimum revient à considérer que les aberrations, les mauvais résultats, viennent de morceaux anormalement bien alignés par le DPLA alors qu'ils n'ont rien à voir. En prenant à chaque fois le minimum des deux scores, on minimiserait donc le nombre de ces aberrations. À l'inverse, si l'on prend le maximum des deux valeurs, on considère chaque analyse comme une tentative d'aligner par DPLA les séquences des morceaux. La composante qui a donné le plus grand score serait donc, dans ce cas, celle qui a trouvé ce qui était similaire entre les deux morceaux.

Nous avons donc essayé ces trois opérations pour fusionner les comparaisons des accompagnements et des mélodies. À l'instar de tous nos tests ultérieurs, les mélodies ont été comparées en utilisant des OTI précalculés sur les morceaux bruts (cf. partie 4.2.2). Les résultats des analyses de morceaux utilisant ces fusions, se trouvent dans le tableau 4.

Pour les trois grandeurs observées, l'opérateur *maximum* obtient les meilleurs résultats. Cependant, ces résultats restent tout de même inférieurs à ceux obtenus avec le DPLA simple sur le morceau brut (cf. tableau 2). Cette infériorité est peut-être due aux mauvais résultats obtenus par les comparaisons sur les solos, comme nous l'avons vu dans la partie 4.2.1. Ces mauvais

	Précision moyenne	Rang moyen	Rang médian
Addition	0,29	36,35	18,5
Minimum	0,20	40,12	22
Maximum	**0,34**	**34,47**	**7**

Tab. 4 – Performances des requêtes pour différentes opérations de fusion à la fin des analyses sur les accompagnements et sur les mélodies

résultats pourraient tirer vers le bas toute analyse faisant intervenir une comparaison des mélodies.

Pour le savoir, nous avons réitéré l'expérience précédente, en remplaçant chaque solo par le morceau brut correspondant. Les résultats, présentés dans le tableau 5, sont bien meilleurs. Dans le cas de l'opérateur *maximum*, les performances sont même meilleures que l'analyse simple sur les morceaux bruts, qui constitue notre état de l'art (cf. page 19). Le rang médian de la reprise passe notamment de 7 à 5. Cela tend à montrer que les solos ont effectivement un mode de comparaison ou de représentation inapproprié.

	Précision moyenne	Rang moyen	Rang médian
Addition	0,34	31,71	7,5
Minimum	0,33	32,66	7
Maximum	**0,37**	**31,40**	**5**

Tab. 5 – Performances des requêtes pour différentes opérations de fusion à la fin des analyses sur les accompagnements et sur les morceaux bruts

4.4.2 Fusion des matrices de similarité

La fusion des analyses à la fin a l'avantage d'être simple à mettre en place mais en fin de compte, on effectue deux analyses séparées. On voudrait voir si les performances sont meilleures en plaçant la jointure plus tôt. En fusionnant les matrices de similarité de l'accompagnement et de la mélodie, on prend en compte plus tôt dans l'analyse, les deux composantes des morceaux. Cela comporte deux avantages. Premièrement, on n'effectuera qu'un seul DPLA pour chaque couple de morceaux, ce qui a l'avantage d'un algorithme moins complexe. Le deuxième avantage est de prendre en compte dès l'alignement par DPLA, les caractéristiques des deux composantes. Ce point pourrait permettre au DPLA d'effectuer des alignements plus pertinents, alignant mieux ensemble les séquences qui se correspondent réellement. Ce système est représenté dans la figure 10.

On commence par calculer deux matrices de similarité (pour la mélodie et pour l'accompagnement), avant de les fusionner. Puis, un DPLA donnera, à partir de cette matrice, un score de similarité.

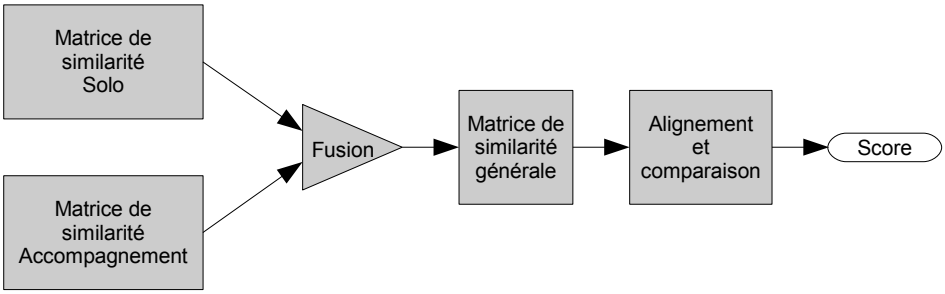

FIG. 10 – Fusion de l'analyse multimodale au niveau des matrices de similarité

Les trois opérateurs de fusion O_n que nous avons testés sur les matrices, sont tous locaux (e.g., chaque point de la matrice résultante dépend des seuls points qui sont situés à la même position dans les deux matrices à fusionner). Soit A et S les matrices de similarité de l'accompagnement et du solo, respectivement. Soit R la matrice résultante (on a donc $R_{i,j} = O_n(i,j)$). Les trois opérateurs sont alors :

1. $O_1(i,j) = \max(A_{i,j}, S_{i,j})$;
2. $O_2(i,j) = \min(A_{i,j}, S_{i,j})$;
3. $O_3(i,j) = \begin{cases} A_{i,j} & \text{si} \quad r_{ij} >= r_m; \\ S_{i,j} & \text{si} \quad r_{ij} < r_m; \end{cases}$

avec $r_{ij} = \frac{P_1(i)+P_2(j)}{2}$, où $P_1(i)$ et $P_2(j)$ représentent, pour chaque morceau, le rapport de puissance du signal entre l'accompagnement et la mélodie aux instants i et j, et $r_m = \frac{\text{mean}_i(P_1(i))+\text{mean}_j(P_2(j))}{2}$.

Comme pour la fusion des scores finaux, prendre chaque maximum équivaut à considérer que la valeur la plus pertinente des deux est celle qui a réussi à trouver des similarités. Au contraire, si on prend le minimum, on juge alors que les valeurs indiquant des similarités sont plus souvent signe d'erreur que les autres. Dans le dernier cas, on se base sur la puissance des signaux : si le rapport entre la puissance de l'accompagnement et du solo, est plus élevé que sa moyenne sur tout le morceau, alors on considère que le soliste ne joue/chante pas. On prend donc la valeur de similarité obtenue sur l'accompagnement. Dans le cas contraire, on prend la valeur de similarité de la mélodie principale.

Nous avons observé les performances des requêtes sur notre base, en utilisant successivement ces trois opérateurs pour fusionner les matrices de similarité de l'accompagnement et de la mélodie principale. Les résultats se trouvent dans le tableau 6. L'opérateur qui effectue un choix d'après la puissance des signaux (O_3) donne les meilleures performances mais l'opérateur *maximum* (O_1) présente des résultats proches.

	Précision moyenne	Rang moyen	Rang médian
O_1	0,26	35,96	**10**
O_2	0,27	42,24	18
O_3	**0,31**	**34,67**	**10**

TAB. 6 – Performances des requêtes en fonction de l'opérateur choisi pour fusioner les matrices de similarité des accompagnements et des solos

Nous avons voulu, comme dans la section précédente, voir si ces systèmes de comparaison de morceaux donneraient de meilleurs résultats en remplaçant la mélodie par le morceau brut. En utilisant le maximum local (O_1), on obtient alors une précision de 0,34, un rang moyen de 30,49, et un rang médian de 5,5. Ces résultats sont bien meilleurs, et même meilleurs que l'état de l'art (cf. page 19).

Ces chiffres semblent confirmer l'efficacité potentielle d'une analyse multimodale, mais les systèmes qui prennent en compte la mélodie principale souffrent encore des mauvaises performances des comparaisons sur ces dernières. Dans la partie 4.5, nous allons tenter de remédier à ce problème en adoptant une autre représentation des mélodies, au niveau de leurs descripteurs.

4.5 Types de descripteurs pour la mélodie principale

Nous avons vu précédemment que nos systèmes d'analyse de similarité musicale étaient détériorés par les mauvaises performances des analyses sur les mélodies principales. Afin de corriger ce défaut, nous avons essayé d'adopter d'autres représentations pour la mélodie.

Les représentations symboliques de la mélodie souffrent souvent d'erreurs d'estimation, qui peuvent détériorer à leur tour les performances du système d'analyse de similarité. C'est pourquoi nous n'avons choisi que des représentations intermédiaires : HPCP standard, HPCP couvrant deux octaves, et produit spectral.

4.5.1 HPCP standard

Les HPCP standard sont les premiers descripteurs que nous avons essayés, puisque ce sont les descripteurs utilisés dans notre système de référence (cf. 3.2.3). Les résultats des analyses utilisant ces descripteurs se trouvent à la page 20. Ces résultats, comme nous l'avons déjà expliqué, nous paraissent insuffisants et appellent donc à trouver un moyen de les améliorer.

4.5.2 HPCP de deux octaves

La nature-même des HPCP les rend incapables de différencier deux octaves différentes. Or dans une mélodie, une différence d'une octave entre

deux notes peut avoir son importance. D'où l'idée de construire des HPCP couvrant deux octaves, qui constituent un intermédiaire entre des HPCP standard et des hauteurs absolues. En effet, ces doubles HPCP sont sensibles aux écarts d'une seule octave mais conservent leur propriété de recouvrement cyclique de l'ensemble des hauteurs.

Calcul des HPCP de deux octaves Le calcul des HPCP standard [6] est décrit en détails dans la section 3.2.2. Il commence par une extraction des pics spectraux, dont la contribution à chaque chroma sera fonction de leur distance avec la fréquence centrale du chroma considéré. Pour un HPCP standard, la fréquence centrale f_n du chroma n est $f_n = f_{\text{ref}} \cdot 2^{\frac{n}{size}}$, où f_{ref} est la fréquence de référence ($f_{\text{ref}} \simeq 440\text{Hz}$) et *size* est la taille du HPCP. Pour nos nouveaux HPCP, il suffit de multiplier n par deux, afin de couvrir deux octaves :

$$f_n = f_{\text{ref}} \cdot 2^{\frac{2n}{size}}$$

avec $n = 1 \ldots size$. Il est important de noter que, si l'on souhaite garder la même résolution que pour des HPCP standard, il est nécessaire de doubler *size*. Ainsi, si on veut par exemple construire un vecteur proposant un chroma par demi-ton, la bonne taille n'est plus 12 mais 24 chroma.

La distance d, en demi-tons, entre f_n et la fréquence f_i de chaque pic i devient :

$$d = 12 \cdot \log_2 \left(\frac{f_i}{f_n} \right) + 24 \cdot m$$

où m est l'entier qui minimise le module de la distance $|d|$.

Mis à part ces changements, la procédure de calcul reste identique à celle décrite dans la section 3.2.2. Un exemple de HPCP de deux octaves se trouve dans la figure 11. Cette séquence décrit une gamme de do majeur allant du do3 (262 Hz) au do5. L'abscisse représente le temps et l'ordonnée représente les classes de hauteurs, dont la résolution a été portée à 120 classes pour la visibilité graphique. On peut distinguer, en noir, les notes jouées par le pianiste.

Transposition Pour calculer l'OTI entre deux mélodies principales, nous avons utilisé l'OTI précalculé sur les morceaux bruts. Mais comme ce dernier a été calculé sur des HPCP d'une seule octave, il est donc défini, pour nos doubles HPCP, à $\frac{size}{2}$ près. Soit o l'OTI précalculé. On a alors :

$$OTI = o + k, \quad k \in \{0, \tfrac{size}{2}\}$$

Nous avons donc, comme pour le calcul d'un OTI classique, calculé la moyenne temporelle de chaque double HPCP sur tout le morceau : h_1 et h_2. Ces deux HPCP globaux sont normalisés par leur valeur maximale. En

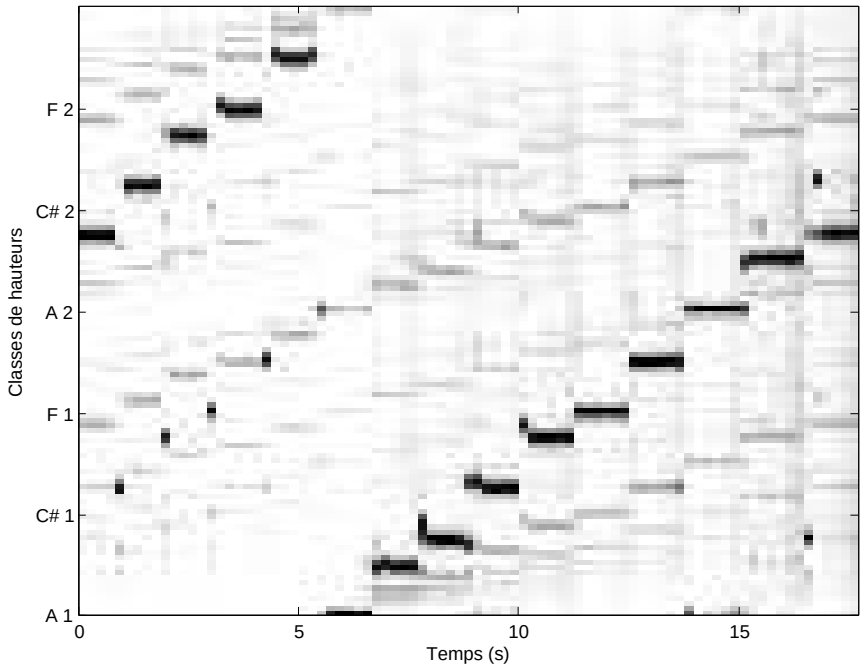

FIG. 11 – Gamme de do majeur sur deux octaves, représentée par des HPCP de deux octaves

adaptant la procédure classique, on va prendre la transposition qui donne le plus grand produit scalaire entre les deux HPCP globaux :

$$OTI(\overrightarrow{h_1}, \overrightarrow{h_2}) = o + \text{argmax}_{k \in \{0, \frac{size}{2}\}} \left\{ \overrightarrow{h_1} \cdot \text{circshift}_R(\overrightarrow{h_2}, o + k) \right\}$$

où "$\cdot$" indique le produit scalaire, et $\text{circshift}_R(\overrightarrow{h}, n)$ est une fonction qui effectue une rotation du vecteur h, de n cases vers la droite.

Test des HPCP de deux octaves Le tableau 7 compare les performances des HPCP d'une et de deux octaves. Les résultats des HPCP de deux octaves, ont été obtenus avec des doubles HPCP de 48 valeurs. Nous avons également fait des tests avec des HPCP de 72 valeurs afin de garder une résolution d'un tiers de demi-ton, mais cela conduit à des performances moins bonnes. De même, en essayant de calculer l'OTI des comparaisons entièrement avec ces HPCP de deux octaves, les performances du système s'en trouvent détériorées.

	Précision moyenne	Rang moyen	Rang médian
HPCP 1 octave	**0,20**	**43,70**	**30**
HPCP 2 octaves	0,18	52,34	41,5

TAB. 7 – Performances de l'analyse de similarité des mélodies avec des HPCP d'une et de deux octaves

Les HPCP de deux octaves ne semblent donc pas adaptés à l'analyse de similarité musicale de mélodies principales.

4.5.3 Produit spectral

Afin, toujours, de tenter de décrire la hauteur tonale d'une mélodie sans utiliser de représentation symbolique, nous avons tenté l'utilisation de séquences de produit spectraux.

Calcul du produit spectral Pour chaque trame d'analyse temps-fréquence, le produit spectral $S(n)$ s'écrit en fonction de la FFT $X(n)$:

$$S(n) = \prod_{m=1}^{M} |X(m \times n)|$$

avec $n < \frac{Nfft}{2M}$.

Les morceaux de notre base Covers80 sont échantillonnés à 16 kHz. En prenant $M = 4$, la fréquence maximale représentée dans notre produit spectral sera donc 2000 Hz. Le produit spectral correspondant à chaque trame d'analyse sera finalement normalisé par sa plus grande valeur.

Transposition du produit spectral Le produit spectral est exprimé sur une échelle linéaire des fréquences. La transposition se fait donc par une multiplication de l'échelle fréquentielle. C'est pourquoi on ne parlera pas ici d'OTI mais d'OTF (*Facteur Optimal de Transposition*). La transposition d'un produit spectral $S(n)$ par un facteur OTF se traduit ainsi :

$$S^t(n) = S(OTF \times n)$$

Puisque $S(n)$ est un signal discret, lorsque $OTF < 1$, les valeurs manquantes seront calculées par interpolation linéaire.

Afin de trouver l'OTF de deux morceaux, on utilise, là encore, un OTI précalculé. Mais à l'instar des HPCP de deux octaves, il reste à déterminer l'octave qui convient. On commence donc par calculer deux vecteurs globaux $\overrightarrow{s_1}$ et $\overrightarrow{s_2}$, en moyennant temporellement les deux séquences de produit spectraux. Puis, $\overrightarrow{s_2}$ sera transposé en fonction de l'OTI précalculé. On va ensuite transposer ce vecteur d'une ou plusieurs octaves, pour trouver l'octave

qui donne le plus grand produit scalaire entre les deux vecteurs. L'octave optimale est donc calculée de la manière suivante :

$$octave = \text{argmax}_{O\in\{-3,-2,\ldots 3\}} \left\{ \overrightarrow{s_1}(n) \cdot \overrightarrow{s_2} \left(n \times 2^{O+\frac{OTI}{hpcpSize}} \right) \right\}$$

où O est le nombre d'octaves par lequel on va élever le produit spectral, "$\cdot$" représente le produit scalaire, OTI est l'OTI précalculé sur les morceaux bruts sans séparation, et *hpcpSize* est la taille des HPCP sur lesquels on a calculé l'OTI.

L'OTF pourra finalement être calculé par :

$$OTF = 2^{octave+\frac{OTI}{hpcpSize}}$$

Notons que nos OTF se sont révélés bien plus pertinents s'ils étaient calculés avant la normalisation des produits spectraux. Nous garderons donc cette manière de procéder.

Similarité du produit spectral Une fois que l'on a transposé nos séquences de produits spectraux, on peut calculer leur matrice de similarité. La distance locale que nous avons choisie entre les instants des morceaux est la distance corrélative (en valeur absolue). Cette distance peut prendre n'importe quelle valeur entre 0 et 1 or nous avons besoin d'une matrice de similarité binaire (cf. section 3.2.3). Il faut donc seuiller les valeurs calculées. La figure 12 montre le rang moyen de la reprise pour un ensemble de requêtes, en fonction du seuil de similarité. La meilleure valeur est obtenue pour un seuil de 0,5, c'est ce seuil que nous garderons.

Test du produit spectral Le tableau 8 montre les performances des analyses de similarité des mélodies basées sur le produit spectral. Ces performances sont en retrait par rapport aux analyses utilisant des HPCP.

	Précision moyenne	Rang moyen	Rang médian
HPCP	**0,20**	**43,70**	**30**
Produits spectraux	0,04	57,15	44,5

TAB. 8 – Performances de l'analyse de similarité des mélodies avec des séquences de HPCP et de produits spectraux

Toutefois, en utilisant cette représentation des mélodies dans nos analyses multimodales, les résultats sont parfois meilleurs que ceux des HPCP. Mais les résultats restent tout de même supérieurs si on remplace nos mélodies en produits spectraux, par les morceaux bruts en HPCP.

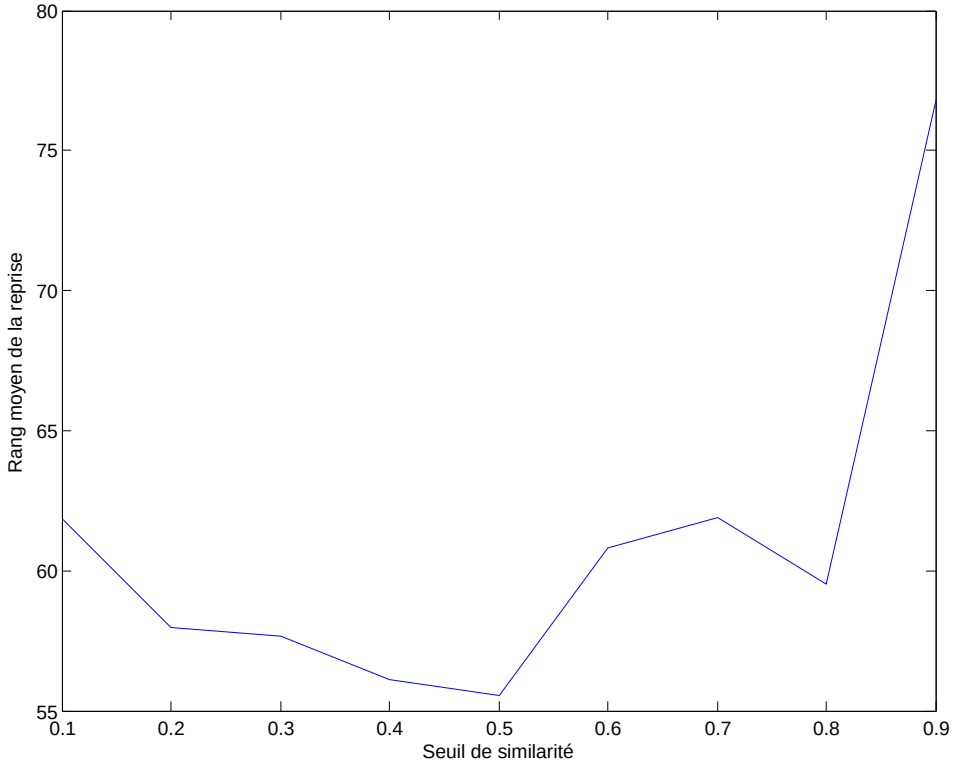

FIG. 12 – Rang moyen de la reprise obtenu, en fonction du seuil de similarité de la corrélation du produit spectral

5 Gestion des silences

Les silences sont certes rares dans les morceaux pop mais certains morceaux contiennent des passages dont la dynamique est très faible. En outre, lors de l'analyse de la mélodie, on est constamment confronté à des passages de durée variable, où le soliste ne joue pas. Ces observations appellent à une gestion réfléchie des silences.

5.1 Pour le morceau brut ou l'accompagnement

Lorsqu'un morceau est joué par un instrument dont le son est non entretenu (comme le piano ou la guitare), la fin de certaines notes a parfois une intensité très faible. Musicalement, on considère que ces notes sont encore jouées mais le signal est perturbé par le bruit de fond. De même, lorsqu'un accompagnement est joué de manière impulsive, "piquée", avec des silences entre toutes les notes, chaque accord résonne dans la mémoire de l'auditeur jusqu'à ce qu'un autre accord vienne le remplacer. Les descripteurs que nous utilisons donnent alors des représentations qui ne collent pas à la réalité et surtout, à l'intention musicale.

Pour remédier à ce problème, nous avons pris un seuil, fixé expérimentalement, en-dessous duquel une trame est considérée silencieuse. Ce seuil a été fixé expérimentalement. Une trame sera ainsi prise pour silencieuse si aucun de ses échantillons n'a une amplitude supérieure à -75 dB par rapport à l'amplitude maximale possible. Les trames silencieuses seront alors remplacées par la trame précédente, afin de simuler la note qui continue. Seul le silence au tout début du morceau sera considéré comme similaire à rien.

Le tableau 9 présente les performances des analyses de similarité effectuées sur les morceaux bruts et sur les accompagnements seuls, avec et sans gestion des silences. Comme on peut le voir, les gains de performances sont minimes mais bien présents.

		Précision moy.	Rang moyen	Rang médian
Brut	Sans gestion	**0,35**	32,02	**7**
	Avec gestion	**0,35**	**31,49**	**7**
Accomp.	Sans gestion	**0,34**	33,18	**7**
	Avec gestion	**0,34**	**33,10**	**7**

Tab. 9 – Performances des analyses de similarité pour les morceaux bruts et les accompagnements seuls, avec et sans gestion des silences

5.2 Pour la mélodie

Dans le cas de la mélodie principale, les silences n'ont pas la même signification que dans l'accompagnement ou le morceau sans séparation. La mélodie n'a ici de réelle existence que lorsque le soliste la joue.

Comme nous l'avons expliqué à la fin de la partie 4.2, lorsque le soliste ne joue/chante pas, notre algorithme d'extraction de la mélodie principale [3] a tendance à extraire, à la place, une partie de l'accompagnement. Cette propriété cause des problèmes lors de la comparaison des solos. Nous avons donc cherché à détecter les trames d'analyse temps-fréquence qui comportaient une partie de l'accompagnement, et à les séparer de celles qui comportaient la mélodie principale.

5.2.1 Similarité des silences

Avant toute chose, il convient de définir un comportement pour les silences, au niveau de la similarité locale. On pourrait considérer que les silences sont tous similaires les uns aux autres puisque, en un sens, ils font partie intégrante de la mélodie. En pratique, cela reviendrait à faire aligner par le DPLA des passages qui n'ont rien à voir les uns avec les autres. Cela donnerait des scores de similarité anormalement élevés à des mélodies comportant beaucoup de silences.

Nous avons plutôt donné aux silences le comportement suivant :
- une trame silencieuse comparée à une trame non silencieuse, donne la valeur "*non similaire*" dans la matrice de similarité ;
- une trame silencieuse comparée à une trame silencieuse, donne une nouvelle valeur, que nous appellerons "*silence*". La matrice de similarité n'est alors plus binaire.

La nouvelle valeur "*silence*" sera ignorée par le DPLA. Lorsque ce dernier tombera sur un "*silence*", la similarité cumulative de la séquence en cours d'alignement n'augmentera ni ne diminuera pas. Cela permet de ne pas pénaliser ni favoriser artificiellement la similarité des mélodies qui comportent beaucoup de silences.

5.2.2 Seuillage simple des silences

Pour trouver une solution aux erreurs d'extraction, nous avons analysé le signal de mélodie, à la sortie de l'algorithme de séparation. Notre analyse a consisté à annoter nous-mêmes quelques morceaux pour distinguer les passages correspondant à la vraie mélodie principale, et les passages comportant des notes issues de l'accompagnement. Nous avons alors constaté que la puissance du signal, en moyenne, était beaucoup plus faible dans les passages erronés que dans ceux qui comportaient la vraie mélodie.

Nous avons alors utilisé un seuil d'énergie en-dessous duquel une trame d'analyse serait considérée comme silencieuse. Ce seuil est fixé séparément pour chaque morceau et dépend de E_m, l'énergie moyenne d'une trame du signal, calculée sur toute la longueur du morceau. La figure 13 montre le rang moyen et médian de la reprise recherchée, pour différentes valeurs du seuil. Les tests ont été effectués avec une représentation sous forme de produit spectral. Nous avons fait ce choix en raison de la rapidité de calcul des produits spectraux, qui nous a permis de tester un plus grand nombre de valeurs. D'après ces résultats, $S = E_m + 5\,\mathrm{dB}$ semble être une bonne valeur pour ce seuil.

Les performances des analyses utilisant ce seuil se trouvent dans le tableau 10 (ces analyses ont été effectuées avec une représentation des mélodies sous forme de HPCP). Malheureusement, les résultats sont moins bons en utilisant le seuil de silence.

	Précision moy.	Rang moyen	Rang médian
Sans seuillage	**0,20**	**43,70**	**30**
Avec seuillage	0,07	54,48	44

TAB. 10 – Performances des analyses de similarité pour la mélodie principale, avec et sans seuillage simple des silences

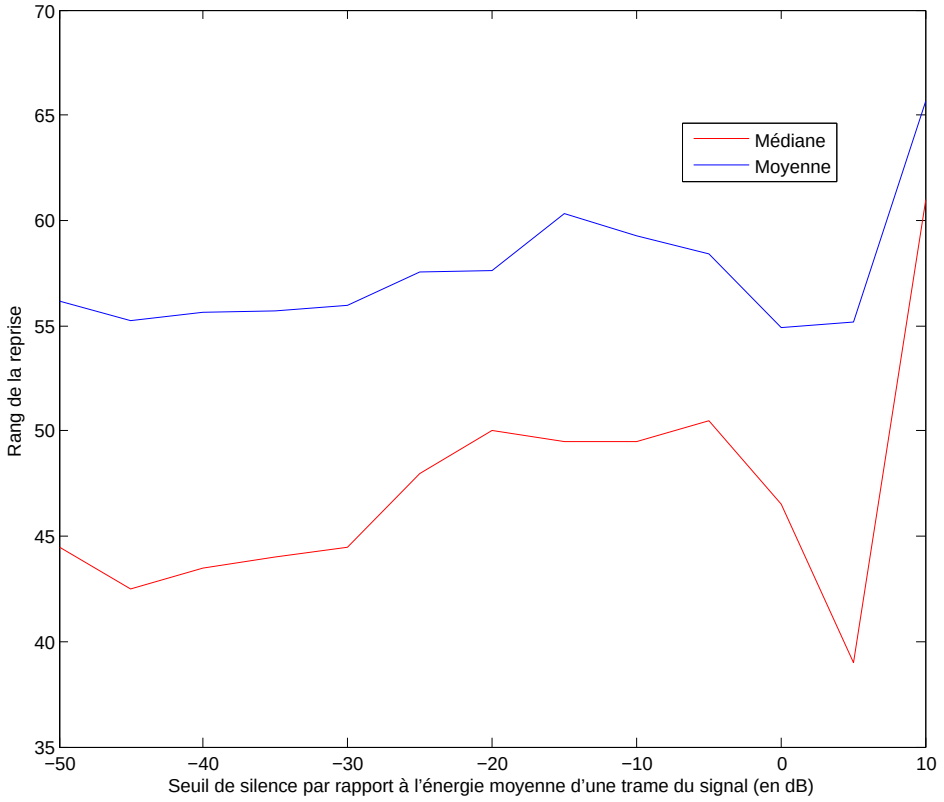

FIG. 13 – Rang moyen et médian de la reprise obtenu lors d'analyses sur les mélodies, en fonction du seuil de silence utilisé

5.2.3 Seuillage des trames silencieuses en tenant compte de la trame précédente

Le seuillage simple des trames permet une détection grossière des silences mais il comporte un défaut : les fins de phrases de la mélodie, souvent en dynamique décroissante, se trouvent souvent en-dessous du seuil d'énergie. Nous avons donc mis en place un nouveau système de détection des silences qui tient compte non seulement de l'énergie de la trame observée, mais également de l'énergie de la trame précédente.

Dans ce système, on compare au seuil fixé, non pas l'énergie de la trame observée, mais sa moyenne pondérée avec l'énergie de la trame précédente. Soit e_i l'énergie de la trame i et S le seuil de silence choisi, une trame sera donc considérée comme silencieuse si

$$\frac{pe_i + (1-p)e_{i-1}}{2} < S$$

avec $0 < p < 1$.

Nous avons testé plusieurs valeurs de p pour plusieurs valeurs de S. La valeur de S qui semble la meilleure est $S = E_m - 30\,\text{dB}$, où E_m est l'énergie moyenne d'une trame dans le morceau considéré. La figure 14 montre l'influence du paramètre p sur les performances de notre système, lorsque $S = E_m - 30\,\text{dB}$. Nous avons gardé la valeur $p = 0.5$ car elle correspond à la fois à la meilleure moyenne, mais aussi à une bonne médiane.

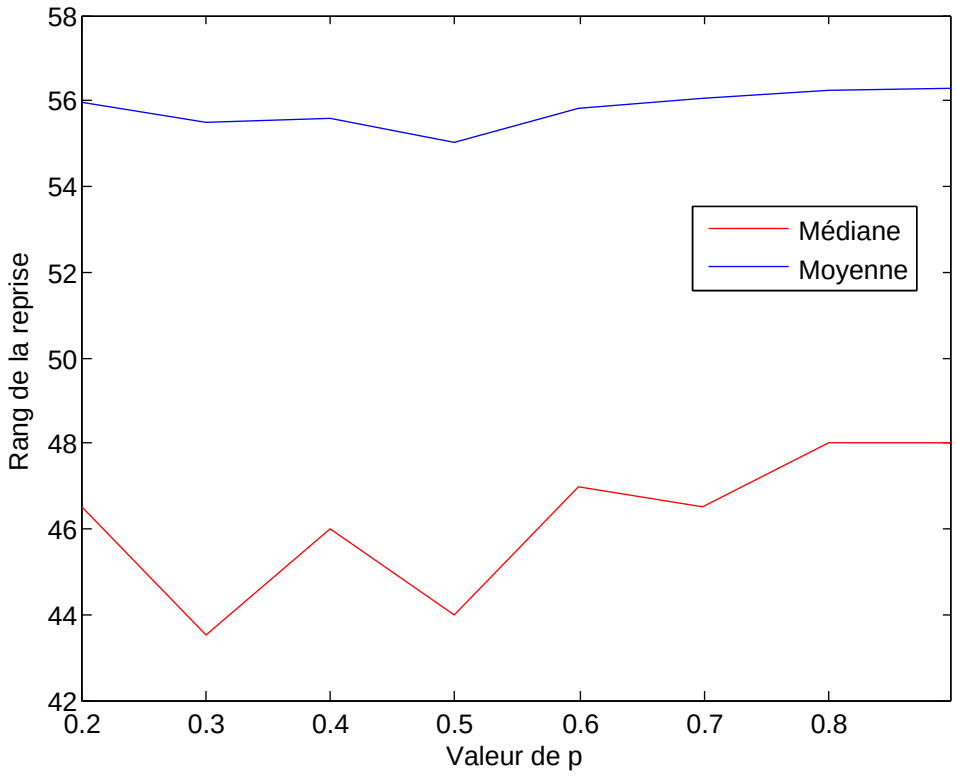

FIG. 14 – Rang moyen de la reprise en fonction la pondération p (pour $S = E_m - 30\,\text{dB}$)

Une fois les valeurs de S et p déterminées, nous avons effectué des tests sur notre base, en utilisant, cette fois, des HPCP. Les résultats se trouvent dans le tableau 11. Si la moyenne du rang de la reprise augmente un peu, la médiane est bien meilleure.

	Précision moyenne	Rang moyen	Rang médian
Sans seuillage	**0,20**	**43,70**	30
Avec seuillage	**0,20**	43,98	**21**

TAB. 11 – Performances de l'analyse de similarité des mélodies, avec et seuillage de l'énergie pondérée

Conclusion

Nous avons vu que l'analyse multimodale des morceaux de musique, considérant séparément la mélodie principale et l'accompagnement, pouvait améliorer la recherche par similarité musicale.

Le système que nous proposons présente des résultats encourageants mais souffre encore des mauvaises performances des analyses basées sur les solos. Ces mauvaises performances admettent trois explications possibles. On pourrait d'abord penser qu'il n'est pas approprié de faire intervenir les mélodies dans ce type d'analyse multimodale. Deuxièmement, le DPLA pourrait se trouver inapproprié à la comparaison de solos. La troisième raison possible serait que les types de descripteurs que nous avons abordés ne sont pas adaptés pour représenter des solos. Cette dernière explication est la plus probable, au vu des écarts de performances observés lors de nos changements de descripteurs. Il faudrait maintenant envisager des approches différentes, comme dans [9], ou encore essayer des descriptions symboliques [17].

Nous avons vu, d'autre part, qu'une gestion réfléchie des silences lors de l'analyse de similarité musicale, que ce soit dans le morceau entier sans séparation, dans l'accompagnement, ou dans la mélodie principale, pouvait également apporter un gain de pertinence. Le gain apporté par cette gestion est léger dans certains cas, mais bel et bien présent. Et le système de gestion des silences que nous proposons demande un temps de calcul négligeable. Toutefois, la question des silences dans la musique est loin d'être simple, et il serait intéressant de l'approfondir, d'un point de vue musicologique ou perceptuel, afin de créer des systèmes plus performants.

Références

[1] S. D. Bella, I. Peretz, and N. Aronoff. Time course of melody recognition : A gating paradigm study. *Percept. Psychophys.*, 7(65) :1019–1028, 2003.

[2] J.-L. Durrieu, G. Richard, and B. David. Singer melody extraction in polyphonic signals using source separation methods. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, pages 169–172, 2008.

[3] J.-L. Durrieu, G. Richard, and B. David. An iterative approach to monaural musical mixture de-soloing. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2009.

[4] D. P. W. Ellis and G. E. Poliner. Identifying cover songs with chroma features and dynamic programming beat tracking. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, volume 4, pages 1429–1432, Apr. 2007.

[5] T. Fujishima. Realtime chord recognition of musical sound : A system using common lisp music. In *Proc. Int. Comput. Music Conf. (ICMC)*, pages 464–467, 1999.

[6] E. Gómez. *Tonal description of music audio signals*. PhD thesis, Music Technol. Group, Univ. Pompeu Fabra, Barcelona, Spain, 2006. Available : http ://www.iua.upf.es/˜egomez/thesis/.

[7] E. Gómez, B. S. Ong, and P. Herrera. Automatic tonal analysis from music summaries for version identification. In *Proc. Audio Eng. Soc. Conv. (AES)*, Oct. 2006.

[8] Eamonn Keogh and Chotirat Ann Ratanamahatana. Exact indexing of dynamic time warping. *Knowl. Inf. Syst.*, 7(3) :358–386, 2005.

[9] M. Marolt. A mid-level melody-based representation for calculating audio similarity. In *Proc. Int. Soc. for Music Inform. Retr. (ISMIR)*, pages 280–285, 2006.

[10] Hendrik Purwins. *Profiles of Pitch Classes — Circularity of Relative Pitch and Key : Experiments, Models, Music Analysis, and Perspectives*. PhD thesis, Technische Universität Berlin, Germany, October 2005.

[11] Lawrence Rabiner and Biing H. Juang. *Fundamentals of Speech Recognition*. Prentice Hall PTR, April 1993.

[12] M. D. Schulkind, R. J. Posner, and D. C. Rubin. Musical features that facilitate melody identification : How do you know it's your song when they finally play it ? *Music Percept.*, 21(2) :217–249, 2003.

[13] J. Serrà. A qualitative assessment of measures for the evaluation of a cover song identification system. In *Proc. Int. Soc. for Music Inform. Retr. (ISMIR)*, pages 319–322, Viena, Austria, 2007.

[14] J. Serrà, E. Gómez, P. Herrera, and X. Serra. Chroma binary similarity and local alignment applied to cover song identification. *IEEE Trans. on Audio, Speech and Language Process.*, 16(6) :1138–1151, Aug. 2008.

[15] A. Sheh and D. P. W. Ellis. Chord segmentation and recognition using em-trained hidden markov models. In *ISMIR*, 2003.

[16] W-H. Tsai, H-M. Yu, and H-M. Wang. A query-by-example technique for retrieving cover versions of popular songs with similar melodies. In *Proc. Int. Soc. for Music Inform. Retr. (ISMIR)*, pages 183–190, 2005.

[17] Rainer Typke, Rainer Typke, Panos Giannopoulos, Panos Giannopoulos, Remco C. Veltkamp, Remco C. Veltkamp, Frans Wiering, Frans Wiering, and Rene Van Oostrum. Using transportation distances for measuring melodic similarity. In *In ISMIR Proceedings*, pages 107–114, 2003.